Artificial neural networks for prediction of the local field potential

Benjamin Latimer[1], Ziao Chen[1], Tyler Banks[1], Dominic Ho[1], Vasiliki Kanta[2], Drew B. Headley[2], Denis Paré[2], Satish S. Nair[1]

[1]Dept. of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO

[2]Center for Molecular and Behavioral Neuroscience, Rutgers University, Newark, NJ

Introduction

- Brain oscillation records reveal a variety of frequencies, with the gamma band (30-80Hz) being prominent in local field potential (LFP) measurements across several brain regions. Interest in brain-machine interfaces and closed-loop stimulator devices that require accurate determination of phase for on-line stimulation feedback has resulted in recent applications of time series techniques to estimate and predict neural signals including the LFP [1-4].
- The electrophysiological origins of the LFP are still being debated (see [5,6] for recent reviews). The unfiltered extracellular measurement is thought to include the sums of action potentials of neurons up to 350 mm from the tip of the micro-electrode, and slower ionic events from within 0.5-3 mm from the tip. This signal is low-pass filtered at ~300 Hz to remove the spike component with the rest termed the LFP. Multiple neuronal processes contribute to the LFP making it difficult to characterize and interpret.
- Although considerable progress has been made, the applications to date have largely involved detection of low frequency components in the LFP signal. Hence, these schemes are not suitable for applications involving higher frequency oscillations such as gamma where nonlinearities and constraints related to implementation make detection of phase for on-line stimulation considerably more challenging. Architectures involving machine learning and hybrid schemes that can be implemented on architectures such as FPGA represent a promising direction for such research.
- We will explore multiple supervised conventional and machine learning approaches to predict *in vivo* LFP recordings 10 ms into the future, using past values. Artificial, convolutional and recurrent neural network (ANN,CNN,RNN) architectures, as well as hybrid versions will be considered. Both raw and filtered (in appropriate bands) versions of the signal will be used. The next step then would be the detection of gamma bursts in the LFP and estimation of it frequency.

Methods

Building the dataset

- For use with a multi-layer perceptron (MLP) learning algorithm, the one dimensional time series data needs to be transformed into an N x M matrix where N is the number of training samples and M is the window size + prediction size.
- We built algorithms for making datasets of variable window lengths and number of samples.
- Two distinct datasets were generated using either the raw signal (Figure 1A) or the signal bandpass filtered between 30-100 Hz (Figure 1B).
- The MLP was implemented in Keras with Tensorflow. It had five hidden layers with 400, 400, 400, 200, and 100 layers, respectively. All used the ReLU activation function and backpropagation for learning.
- 80% of the data was used for training and 20% were retained for testing. The results shown are for testing (unseen) data, for illustrating the network's ability to generalize from the training data.
- We compared the root mean squared error to the "persistence forecast" to judge the effectiveness of the network. The persistence forecast is using the last sample of the input as the one and only value for the predicted samples.

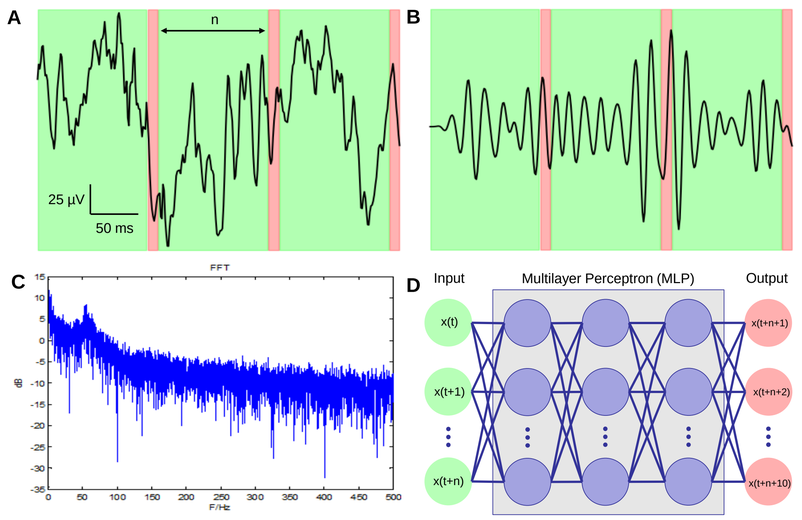

Figure 1. Dataset and machine learning approach.
A: Segment of an LFP trace. Green show the samples considered as input and red show the output samples. **B:** The same LFP trace in A, bandpass filtered at 70-100 Hz. **C:** FFT of signal, showing clear bump in gamma band (~60Hz). **D:** Cartoon of multi-layer perceptron used for learning and predicting.

Results & Discussion

Linear modeling: Autoregressive Integrated Moving Average (ARIMA)

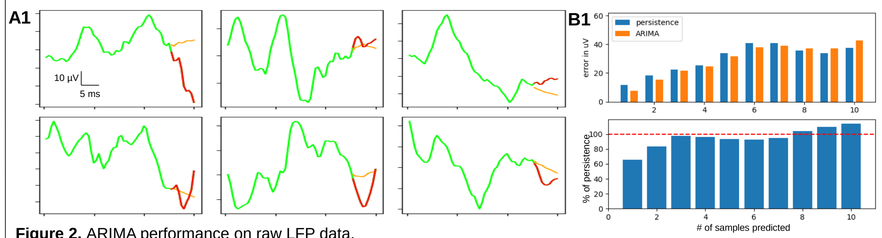

Figure 2. ARIMA performance on raw LFP data.
A: Six examples of the MLP predictions on unseen data. Green shows the input time samples, red shows the true output, and orange is the prediction. **B:** Bar graph of error in prediction for persistence and MLP.

- The linear model (p, q, d = 5, 1, 0) did not perform better than the persistence forecast.

MLP predicting on raw LFP

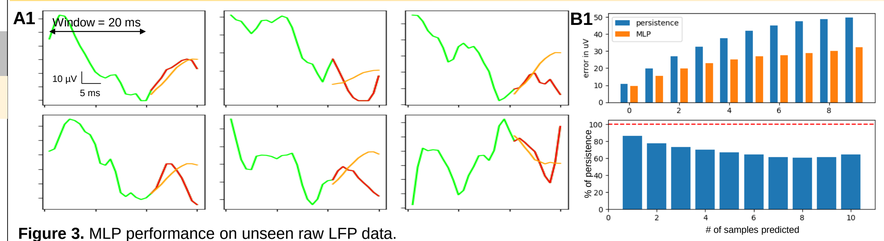

Figure 3. MLP performance on unseen raw LFP data.
A1-3: Six examples of the MLP predictions on unseen raw LFP data for a window size of 25 (A1, B1), 10 (A2, B2), and 5 (A3,B3). Green shows the input time samples, red shows the true output, and orange is the prediction. **B1-3:** Bar graph of error in prediction for persistence and MLP.

- Testing on the raw LFP signal yielded a root mean squared error (RMSE) of between 10.1 and 40.2 µV, depending on the size of the input window. Various input window sizes were tried, with 5 ms being optimal (Figure 3B3).

MLP predicting on causal filtered LFP

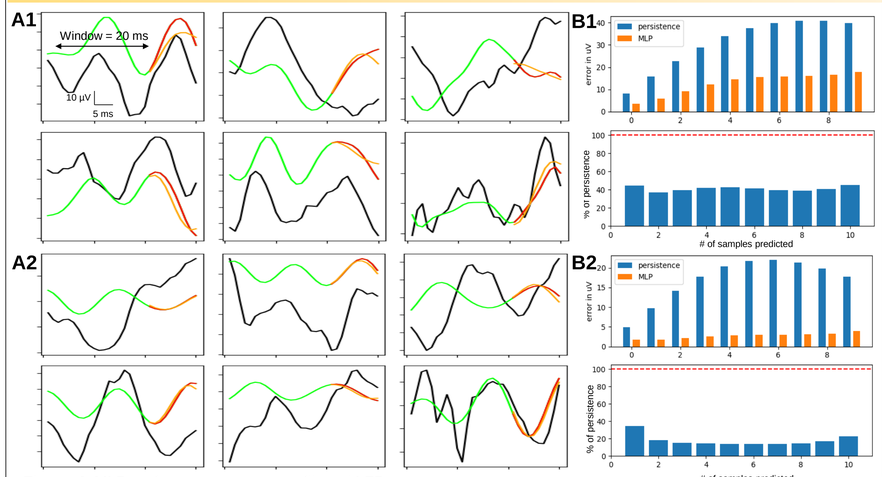

Figure 4. MLP performance on unseen filtered LFP data.
A1-2: Six examples of the MLP predictions on unseen filtered (30-100 Hz – A1, B1; 60-80 Hz – A2, B2) data. Black is the raw LFP trace. Green shows the input time samples, red shows the true output, and orange is the prediction. **B1-2:** Bar graph of error in prediction for persistence and MLP.

- Testing on the band-pass filtered signal yielded a root mean squared error (RMSE) of between 1.8 and 5.2 µV, compared to an error with the persistence forecast of 4.9 and 12.5 µV. The MLP RMSE was consistently less than 50% and as much as 80% better than the persistence forecast, especially when we filtered on a narrow band (Fig. A2-B2.
- To arrive at this network, we varied the number of nodes, hidden layers, and window size (nodes in the input layer). We found the optimal window size was 20 ms.

Results & Discussion

LSTM predicting on raw LFP

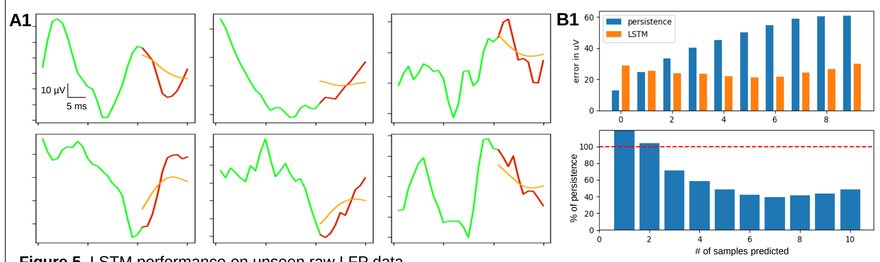

Figure 5. LSTM performance on unseen raw LFP data.
A1: Six examples of the LSTM predictions on unseen raw LFP data. Green shows the input time samples, red shows the true output, and orange is the prediction. **B1:** Bar graph of error in prediction for persistence and LSTM.

- Long Short-Term Memory (LSTM) networks have been shown to be useful in time series prediction [7]. We used a network of two layers of 200 LSTM units each. We used walk-forward validation whereby we made one prediction and then incorporated that prediction into the network in order to make another prediction.
- The dataset was the same as in the MLP studies except we simultaneously fed filtered and raw data to the network to predict the raw samples. We used 4 filter bands: a low-pass filter below 5 Hz, bandpass between 5-30 and 30-100 Hz, and a highpass filter above 100 Hz. Together with the raw signal, this made five inputs to the LSTM network.
- Compared to the MLP, the LSTM was better at forecasting on the raw signal in some cases. At the 10th prediction, the RMSE was 40% of the persistence compared to 60% for the MLP.
- LSTM performance on the filtered signal was comparable to the MLP so we didn't include it here.

Discussion and future directions

- Prediction of local field potential signals is a challenging task, especially in this dataset where the prominent oscillation (gamma) is sporadic - it is often not sustained for more than a few tens of milliseconds.
- Given that our neural network approach worked very well on filtered data, it could be that LFP signals with greater periodicity would be easier to predict.
- The long short-term memory network proved to be skillful in predicting the raw signal, compared to the multi-layer perceptron. Since the LSTM network accepts multiple channels, it could be that using signals from multiple electrodes as inputs to predict the signal of one electrode could increase the accuracy. This of course assumes that the signals are somehow dependent on one another – a reasonable assumption in brain signals that are anatomically close by.
- After increasing the accuracy of the MLP or LSTM, the next step will be implementing these algorithms for real-time closed-loop stimulation. Neural networks have been implemented in Field-Programmable Gate Arrays (FPGAs) allowing for fast computation and stimulation [8].
- We will eventually work toward making this a general framework that could be applied to any LFP signal for use in closed-loop stimulation applications.
- All code is open-source and is available at github.com/latimerb/LFP_Prediction.

References

[1] W.S. Anderson, P. Kudela, S. Weinberg, G.K. Bergey, P.J. Franaszczuk, Phase-dependent stimulation effects on bursting activity in a neural network cortical simulation, Epilepsy Res, 84 (2009) 42-55.
[2] L.L. Chen, R. Madhavan, B.I. Rapoport, W.S. Anderson, Real-Time Brain Oscillation Detection and Phase-Locked Stimulation Using Autoregressive Spectral Estimation and Time-Series Forward Prediction, IEEE Trans. Biomed. Eng., 60 (2013) 753-762.
[3] C. Brunner, R. Scherer, B. Graimann, G. Supp, G. Pfurtscheller, Online control of a brain-computer interface using phase synchronization, IEEE Trans. Biomed. Eng., 53 (2006) 2501-2506.
[4] T.C. Marzullo, M.J. Lehmkuhle, G.J. Gage, D.R. Kipke, Development of closed-loop neural interface technology in a rat model: combining motor cortex operant conditioning with visual cortex microstimulation, IEEE Trans Neural Syst Rehabil Eng, 18 (2010) 117-126.
[5] G.T. Einevoll, C. Kayser, N.K. Logothetis, S. Panzeri, Modelling and analysis of local field potentials for studying the function of cortical circuits, Nat. Rev. Neurosci., 14 (2013) 770-785.
[6] B. Pesaran, M. Vinck, G.T. Einevoll, A. Sirota, P. Fries, M. Siegel, W. Truccolo, C.E. Schroeder, R. Srinivasan, Investigating large-scale brain dynamics using field potential recordings: analysis and interpretation, Nat. Neurosci., 21 (2018) 903-919.
[7] S. Hochreiter and J. Scmidhuber. Long Short-Term Memory, Neural Computation, 9(8) (1997) 1735-1780.
[8] A.X.M. Chang, B. Martini, and E. Culurciello. (2016). Recurrent neural networks hardware implementation on FPGA. arXiv:1511.05552 [cs.NE]

Acknowledgments

This work was supported by grants NSF OAC-1730655, and NIMH grants MH087755 & MH109122.